

TP4 : Base de données Relationnelles Répartie : Fragmentation Horizontale

Objectifs :

- Fragmentation d'une base de données relationnelles en respectant un ensemble de critère de sélection.
- Utilisation des requêtes SQL des triggers des procédures stockées PL/SQL pour gérer les données distribuées.

Exo-1

Soit la base de données relationnelles suivantes stockée dans le site BANK:

Client(NoClient, NomClient, PrénomClient, VilleClient, Age)

Agence(CodeAgence, NomAgence, Adresse, Ville)

Compte(IdCompte, NCompte, Solde, NoClient, CodeAgence)

Travail à faire :

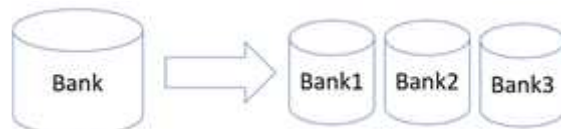
Soient les requêtes de sélection les plus utilisées dans trois sites différents BANK1, BANK2 et BANK3 :

$$R_1 = \sigma_{VilleClient='Casablanca' \text{ AND } AGE < 40}(Clients)$$

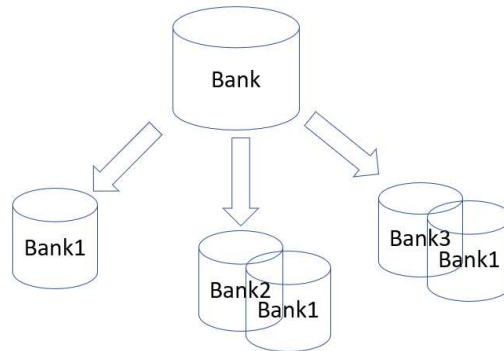
$$R_2 = \sigma_{VilleClient='Rabat' \text{ AND } Solde < 0}(Clients \bowtie Comptes)$$

$$R_3 = \sigma_{VilleClient='Rabat' \text{ AND } Solde \geq 0}(Clients \bowtie Comptes)$$

- 1- A l'aides des requêtes SQL exécutées dans BANK, implémenter les schémas fragmentés de BANK dans les trois comptes (Sites) BANK1, BANK2, BANK3. (la fragmentation concernera que les tables clients et comptes).



- 2- Dupliquer le fragment de la Bank1 dans les BDD de bank2 et bank3.



- 3- Créer deux triggers « SYNCHRONE_CLIENTS » et « SYNCHRONE_COMPTES » qui permettent de synchroniser les mises à jour dans les fragments dupliqués de Bank1.
- 4- Créer un trigger qui permet de synchroniser la répartition des insertions des nouveaux comptes dans la BDD global « Bank » vers les fragments concernés par cette insertion.
- 5- Créer les triggers permettant de synchroniser la répartition des Updates dans les tables « client » et « compte » vers les fragments concernés.
- 6- Créer les triggers permettant de synchroniser la répartition les suppressions dans les tables « client » et « compte » vers les fragments concernés.